Министерство образования Республики Беларусь

Учреждение образования «Брестский государственный технический университет»

Кафедра ИИТ

Отчёт по лабораторной работе №7

Выполнил:

студент 3 курса

Группы ПО-5

Поздняков Д.А.

Проверил:

Крощенко А.А.

Брест 2021

**Цель работы:** освоить возможности языка программирования C# в построении графических приложений.

**Вариант 5**

**Задание 1.** Изобразить в окне приложения (апплета) отрезок, вращающийся в плоскости экрана вокруг одной из своих концевых точек. Цвет прямой должен изменяться при переходе от одного положения к другому.

**Задание 2.** Реализовать построение фрактала Дерево Пифагора.

**Код программы (задание 1):**

Form1.cs

using System;

using System.Drawing;

using System.Windows.Forms;

namespace spp.lab7

{

public partial class Form1 : Form

{

private Segment segment;

public Form1()

{

InitializeComponent();

segment = new Segment(100, 100, 160, 180);

var bt = new Button() { Parent = this, Text = "Rotate" };

bt.Click += bt\_Click;

}

void bt\_Click(object sender, EventArgs e)

{

double Angle = 15 \* Math.PI / 180f;

segment.Rotate(Angle);

Invalidate();

}

protected override void OnPaint(PaintEventArgs e)

{

Pen red = new Pen(Color.Red, 5);

Pen orange = new Pen(Color.Orange, 5);

Pen yellow = new Pen(Color.Yellow, 5);

Pen green = new Pen(Color.Green, 5);

Pen lightBlue = new Pen(Color.LightBlue, 5);

Pen blue = new Pen(Color.Blue, 5);

Pen magenta = new Pen(Color.Magenta, 5);

if(segment.End.X > 0 && segment.End.X < 10)

e.Graphics.DrawLine(red, segment.Start, segment.End);

if (segment.End.X > 10 && segment.End.X < 45)

e.Graphics.DrawLine(orange, segment.Start, segment.End);

if (segment.End.X > 45 && segment.End.X < 80)

e.Graphics.DrawLine(yellow, segment.Start, segment.End);

if (segment.End.X > 80 && segment.End.X < 115)

e.Graphics.DrawLine(green, segment.Start, segment.End);

if (segment.End.X > 115 && segment.End.X < 150)

e.Graphics.DrawLine(lightBlue, segment.Start, segment.End);

if (segment.End.X > 150 && segment.End.X < 190)

e.Graphics.DrawLine(blue, segment.Start, segment.End);

if (segment.End.X > 190 && segment.End.X < 200)

e.Graphics.DrawLine(magenta, segment.Start, segment.End);

}

}

class Segment

{

public PointF Start;

public PointF End;

public Segment(float x1, float y1, float x2, float y2)

{

Start = new PointF(x1, y1);

End = new PointF(x2, y2);

}

public void Rotate(double angle)

{

var pp = new PointF(End.X - Start.X, End.Y - Start.Y);

var x = pp.X \* Math.Cos(angle) - pp.Y \* Math.Sin(angle);

var y = pp.X \* Math.Sin(angle) + pp.Y \* Math.Cos(angle);

End = new PointF((float)x + Start.X, (float)y + Start.Y);

}

}

}

Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace spp.lab7

{

static class Program

{

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

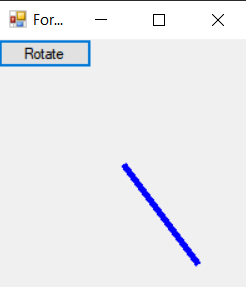
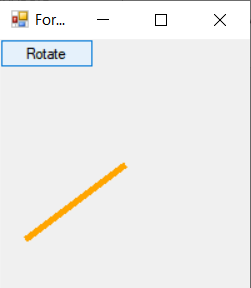
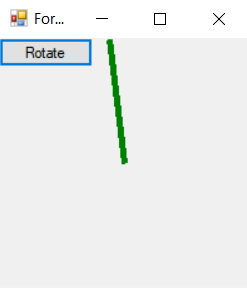
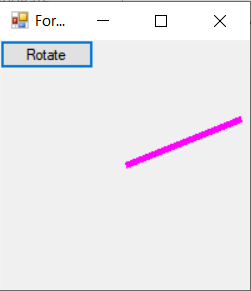
Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

**Код программы (задание 2):**

Form1.cs

using System;

using System.Drawing;

using System.Windows.Forms;

namespace PythagorasTree

{

public partial class Form1 : Form

{

public Graphics g;

public Bitmap map;

public Pen p;

public double angle = Math.PI / 2; //Угол поворота на 90 градусов

public double ang1 = Math.PI / 4; //Угол поворота на 45 градусов

public double ang2 = Math.PI / 6; //Угол поворота на 30 градусов

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

map = new Bitmap(pictureBox1.Width, pictureBox1.Height);

g = Graphics.FromImage(map);

g.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.AntiAlias;

p = new Pen(Color.Black);

DrawTree(300, 450, 200, angle);

pictureBox1.BackgroundImage = map;

}

//Рекурсивная функция отрисовки дерева

//x и y - координаты родительской вершины

//a - кол-во итераций

//angle - угол поворота на каждой итерации

public int DrawTree(double x, double y, double a, double angle)

{

if (a > 2)

{

a \*= 0.7;

double xnew = Math.Round(x + a \* Math.Cos(angle)),

ynew = Math.Round(y - a \* Math.Sin(angle));

g.DrawLine(p, (float)x, (float)y, (float)xnew, (float)ynew);

x = xnew;

y = ynew;

DrawTree(x, y, a, angle + ang1);

DrawTree(x, y, a, angle - ang2);

}

return 0;

}

}

}

Program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace PythagorasTree

{

static class Program

{

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

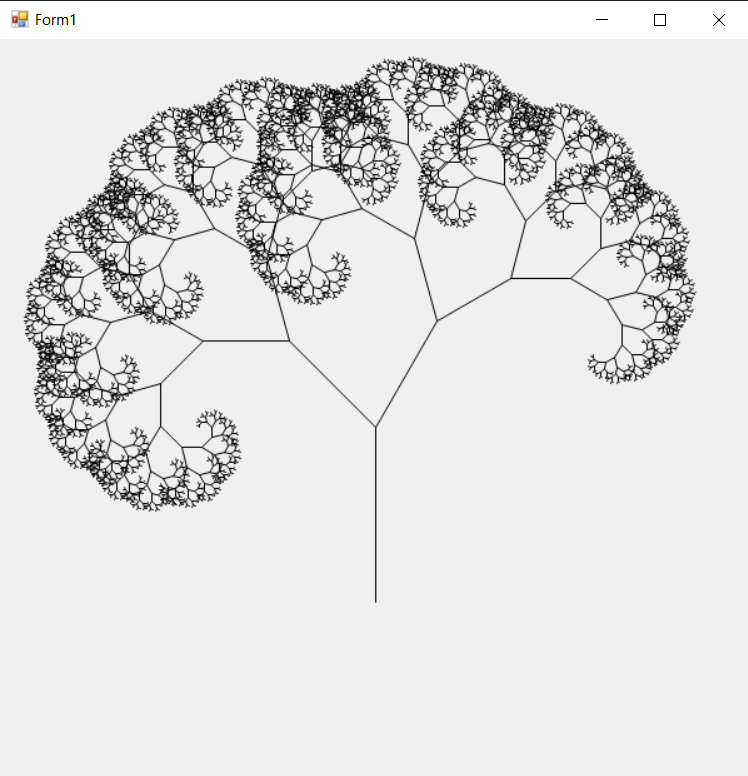
Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}



Вывод: в ходе выполнения лабораторной работы освоил возможности языка программирования C# в построении графических приложений.